Clustering for Personalized Preference Prediction

Fan Yang[*] Xiaotong Shen[†]

Abstract

We build a model to allow personalized prediction for different individuals on a large amount of items based on both user features and item features, as in a recommender system. User and item "preferences" are clustered through supervised learning by modeling the observed response with a gaussian distributed regression model. Besides mean parameters, correlation structure of the response variable is also modeled. Fusion type penalties are applied to identify similar users and items. Simulation results show our model performs better than the popular matrix decomposition methods.

Key Words: Clustering, Personalized, Recommendation

1. Introduction

Recent years have seen a wide application of recommender systems. For example, Amazon recommends items for customers who browse their website, Netflix recommends movies to its users, Expedia make recommendations about flight and hotels based on customer history behavior and so on. Given users' past preference on the consumed items, recommender system predict the preferences of users on unseen items.

The earlist recommender system is through collaborative filtering developed in the mid-1990s [1]. There are a lot variants of SVD decomposition type of methods proposed. These methods use latent variables to represent users and items. But in reality, in many cases co-variates of users or items are available. For example, the demographic information of users may be known like gender, occupation, age, e.t.c.; for an item, some content information may be known like the genre of a movie, the category of a product, the price of a hotel e.t.c. The existing methods don't take these covariate information into consideration, which are likely to be very useful in predicting the behavior of users on the items. Our model utilizes the available user and item feature information. For each user and item, we estimate their individual "preference" on the item feature and user feature respectively. We assume the user and item preferences form clusters and add fusion type of penalty to estimate it. More-over, since the ratings on different items given by a single user is likely to be correlated, we estimate the correlation structure between ratings given by one user. So our model is a personalized recommender system with clustering structure and correlation struture. The details of our model will be given in section 2.

2. Proposed Models

2.1 Models

Consider a situation where we have an $n \times m$ rating matrix $\boldsymbol{R} = (r_{ij})_{n\times m}$, with each row and column corresponding to one user and one item, and r_{ij} is the rating of user i on movie j. Some entries of $\boldsymbol{R}$ may be missing. To account for correlations among ratings on items associated with the same user, we assume that ratings from a user follow a multivariate

[*] School of Statistics, University of Minnesota, Minneapolis, MN 55455.

[†] School of Statistics, University of Minnesota, Minneapolis, MN 55455.

normal distribution with some covariance matrix, whereas ratings from different users are independent.

To be specific, suppose user i rates m_i items with index set $I_i \triangleq \{i_1, i_2, \cdots, i_{m_i}\} \subseteq \{1, 2, \cdots, m\}$, where $i_1 < i_2 < \cdots < i_{m_i}$. For observed ratings $\boldsymbol{r}_i = (r_{i,i_1}, r_{i,i_2}, \cdots, r_{i,i_{m_i}})^T$ from user i, we assume $\boldsymbol{r}_i \sim N(\boldsymbol{\mu}_i, \boldsymbol{\Omega}_i^{-1})$, where $\boldsymbol{\mu}_i = (\mu_{i,i_1}, \mu_{i,i_2}, \cdots, \mu_{i,i_{m_i}})^T$ is the mean of the observed ratings from user i, and $\boldsymbol{\Omega}_i$ is the precision matrix to describe the correlations of observed ratings from user i. Here the precision matrix is used as opposed to the covariance matrix to facilitate computation, because the log likelihood is convex in $(\boldsymbol{\Omega}_1, \cdots, \boldsymbol{\Omega}_n)$ but not in the covariance matrix. More formally, our prediction model can be written as

$$r_{ij} = \mu_{ij} + \epsilon_{ij}, \quad \mu_{ij} = \boldsymbol{x}_i^T \boldsymbol{\alpha}_j + \boldsymbol{y}_j^T \boldsymbol{\beta}_i, \quad (\epsilon_{i,i_1}, \cdots, \epsilon_{i,i_{m_i}})^T \sim N(\boldsymbol{0}, \boldsymbol{\Omega}_i^{-1}); \tag{1}$$

$i = 1, 2 \cdots, n$, $j = i_1, i_2, \cdots, i_{m_i}$, where $\boldsymbol{x}_i$ is user i feature vector such as demographic information, $\boldsymbol{y}_j$ is item j feature vector such as genre of movie j, $\boldsymbol{\alpha}_j$ is a vector representing "preference" of item j over user feature variables, $\boldsymbol{\beta}_i$ is a vector representing "preference" of user i over item feature variables, and ϵ_{ij} is the random error. Let $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, \cdots, \boldsymbol{\alpha}_m)$, $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \cdots, \boldsymbol{\beta}_n)$, $\boldsymbol{\Omega} = (\boldsymbol{\Omega}_1, \cdots, \boldsymbol{\Omega}_n)$. The log-likelihood can be written as

$$l(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Omega}) = \sum_{i=1}^{n} \left[\frac{1}{2} \text{logdet}(\boldsymbol{\Omega}_i) - \frac{(\boldsymbol{r}_i - \boldsymbol{\mu}_i)^T \boldsymbol{\Omega}_i (\boldsymbol{r}_i - \boldsymbol{\mu}_i)}{2} \right]. \tag{2}$$

To identify user and item clusters, we penalize the pairwise differences among $\boldsymbol{\alpha}_j$'s and $\boldsymbol{\beta}_i$'s. For the precision matrix, an $m \times m$ matrix $\boldsymbol{\Omega}_{T_i}$ is estimated for all i. The submatrix of $\boldsymbol{\Omega}_{T_i}$ with row and column indices for items rated by user i is $\boldsymbol{\Omega}_i$. If item k or item pair (k, l) is rated by at least one user, pairwise differences of $\boldsymbol{\Omega}_{T_i}$ entries $\omega_{T_i,kk}$ or $\omega_{T_i,kl}$ across different i's are penalized. The (k, l) entry of $\boldsymbol{\Omega}_{T_i}$ for all i is fixed at 0 if no user rated this pair. Specifically, for item pair (k, l) with $k < l$, suppose it is rated by at least one user. Then we penalize the difference $|\omega_{T_i,kl} - \omega_{T_j,kl}|$ for all $i \neq j$. The diagonal difference $|\omega_{T_i,kk} - \omega_{T_j,kk}|$ is also penalized for all item k and all $i \neq j$. Let $\boldsymbol{S}_i = (\boldsymbol{r_i} - \boldsymbol{\mu_i})(\boldsymbol{r_i} - \boldsymbol{\mu_i})^T$ and J be a general penalty function, the penalized log likelihood is

$$\begin{aligned} l(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Omega}) = & \frac{1}{2} \sum_{i} [\text{logdet}(\boldsymbol{\Omega}_i) - \text{tr}(\boldsymbol{\Omega}_i \boldsymbol{S}_i)] - \frac{\lambda_1}{2} \sum_{i<j} \sum_{t} J(|\alpha_{it} - \alpha_{jt}|) \\ & - \frac{\lambda_1}{2} \sum_{i<j} \sum_{t} J(|\beta_{it} - \beta_{jt}|) - \lambda_2 \sum_{i<j} \sum_{k \leqslant l, \{k,l\} \subseteq \bigcup_{h=1}^{n} I_h} J(|\omega_{T_i,kl} - \omega_{T_j,kl}|), \end{aligned} \tag{3}$$

where $\lambda_1, \lambda_2 > 0$ are regularization parameters. We maximize (3) with respect to $\boldsymbol{\alpha}_i$'s, $\boldsymbol{\beta}_i$'s and $\boldsymbol{\Omega}_{T_i}$'s.

For the penalty function J, we considered two specific forms, the L_1-norm and the non-convex truncated L_1 penalty [11], denoted as TLP. Techniques used for the L1 regularized objective are the same as for minimizing the TLP regularized objective except the difference of convex method is not used. So we only give the details about solving the problem with TLP regularization. The TLP function is defined as $J_\tau(x) = \min(|x|, \tau)$, where $J_\tau(x)/\tau$ approximates the L_0-penalty as $\tau > 0$ goes to 0^+. Details for solving this problem will be given in the next section.

2.2 Algorithm

To minimize (3) with TLP, which is non-convex, we combine the difference of convex algorithm, blockwise coordinate descent algorithm, alternating direction method of multipliers algorithm, and alternating minimization algorithm [3] to solve a convex relaxation of it. We update the mean and the precision matrix alternately.

A DC decomposition of J_τ is $J_\tau(x) = |x| - \max(|x| - \tau, 0)$. By using this decomposition we approximate the nonconvex TLP with a convex minorization iteratively.

First, we apply the d.o.c. at the outermost loop. Then in each iteration of the d.o.c., we use blockwise coordinate descent with $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Omega}$ as the blocks. To deal with the fusion type penalty on $\boldsymbol{\alpha}, \boldsymbol{\beta}$ and $\boldsymbol{\Omega}$, we use the alternating direction method of multipliers. The alternating minimization algorithm is applied when we solve for the proxy variable for $\boldsymbol{\Omega}$.

3. Simulation Result

We did simulation of our models on 100 users, 30 items. The missing proportion of the ratings is set at 0.8 for all users. The dataset is divided 3:1:1 for train, tune and test. We compared four methods: (a.) the SOFT-IMPUTE in [8] which penalize the nuclear norm in matrix completion, (b.) the special case of our proposed model which doesn't consider precision matrix (fix at identity matrix) with L_1-norm clustering, (c.) our proposed model with L_1-norm clustering and (d.) our proposed model with TLP clustering. The tuning parameters tried in the four models are grid values. We considered four choices of $\boldsymbol{\Omega}$: s1 and s2 are two scales, m1/m2 uses dependent/independent errors between train, tune, test.

To compare the performance of different methods, the root mean squared error (RMSE) and weighted root mean squared error (wRMSE) on the test set are calculated. The wRMSE uses the true test set precision matrix to weight the errors.

The simulation result is as shown in the following figure. Our methods compares favorably to the SOFT-IMPUTE matrix completion method (about 50% improvement). Modeling correlation structure shows advantage than ignoring it under settings m1 (7.5%-10% improvement).

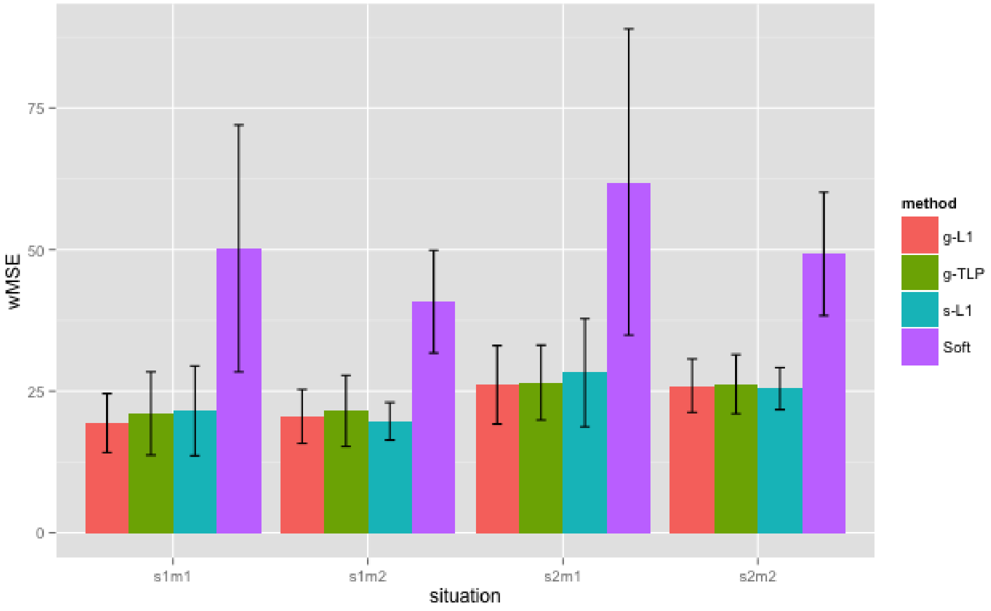

Figure 1: Simulation Results of Four Methods with SEs

References

[1] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions On Knowledge And Data Engineering*, 17(6):734–749, 2005.

[2] Robin Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.

[3] Eric C. Chi and Kenneth Lange. Splitting methods for convex clustering. *Journal of Computational and Graphical Statistics*, 23(1):111–128, 2014.

[4] Patrick Danaher, Pei Wang, and Daniela M. Witten. The joint graphical lasso for inverse covariance estimation across multiple classes. *Journal of the Royal Statistical Society: Series B*, 76(2):373–397, 2014.

[5] Rina Foygel, Nathan Srebro, and Ruslan Salakhutdinov. Matrix reconstruction with the local max norm. In *NIPS*, pages 944–952, 2012.

[6] Holger Hoefling. A path algorithm for the fused lasso signal approximator. *Journal of Computational and Graphical Statistics*, 19(4):984–1006, 2010.

[7] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems, 2009.

[8] Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *Journal of Machine Learning Research*, 11, 2010.

[9] Jennifer Nguyen and Mu Zhu. Content-boosted matrix factorization techniques for recommender systems. *Statistical Analysis and Data Mining*, 6(4):286 – 301, 2013.

[10] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. Grouplens: An open architecture for collaborative filtering of netnews. pages 175–186. ACM Press, 1994.

[11] Xiaotong Shen, Wei Pan, and Yunzhang Zhu. Likelihood-based selection and sharp parameter estimation. *Journal of American Statistical Association*, 107:223–232, 2012.